FIG.1

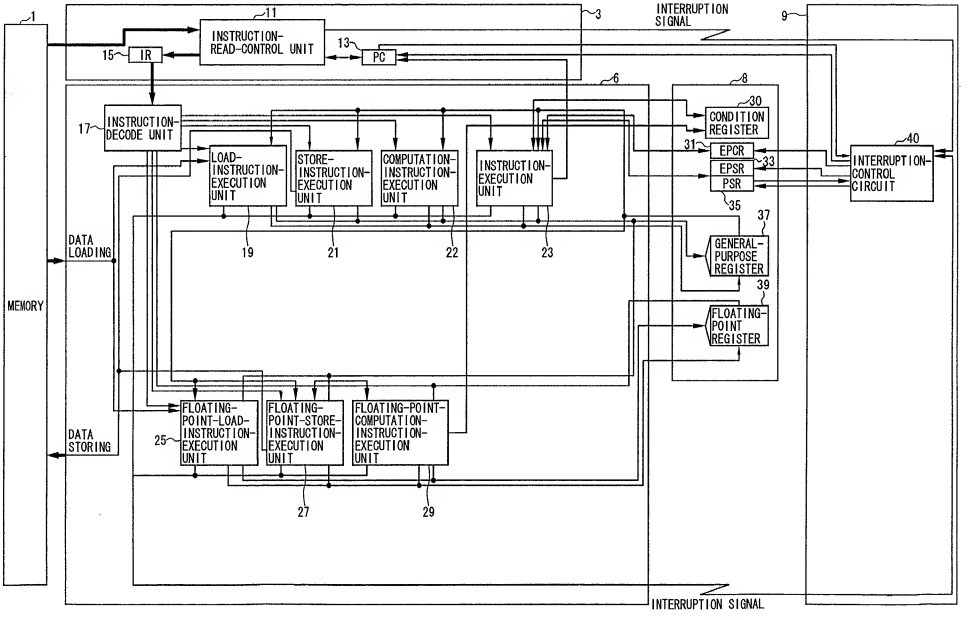

1
MEMORY
11
INSTRUCTION-READ-CONTROL UNIT
3
INTERRUPTION SIGNAL
9
15
IR
13
PC
6
8
17
INSTRUCTION-DECODE UNIT
30
CONDITION REGISTER
31
EPCR
33
EPSR
PSR
35
40
INTERRUPTION-CONTROL CIRCUIT
LOAD-INSTRUCTION-EXECUTION UNIT
STORE-INSTRUCTION-EXECUTION UNIT
COMPUTATION-INSTRUCTION-EXECUTION UNIT
INSTRUCTION-EXECUTION UNIT
19
21
22
23
37
GENERAL-PURPOSE REGISTER
39
FLOATING-POINT REGISTER
DATA LOADING
DATA STORING
25
FLOATING-POINT-LOAD-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-STORE-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-COMPUTATION-INSTRUCTION-EXECUTION UNIT
27
29
INTERRUPTION SIGNAL

FIG.2

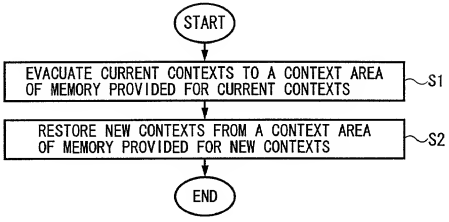

START
EVACUATE CURRENT CONTEXTS TO A CONTEXT AREA OF MEMORY PROVIDED FOR CURRENT CONTEXTS
S1
RESTORE NEW CONTEXTS FROM A CONTEXT AREA OF MEMORY PROVIDED FOR NEW CONTEXTS
S2
END

FIG.3

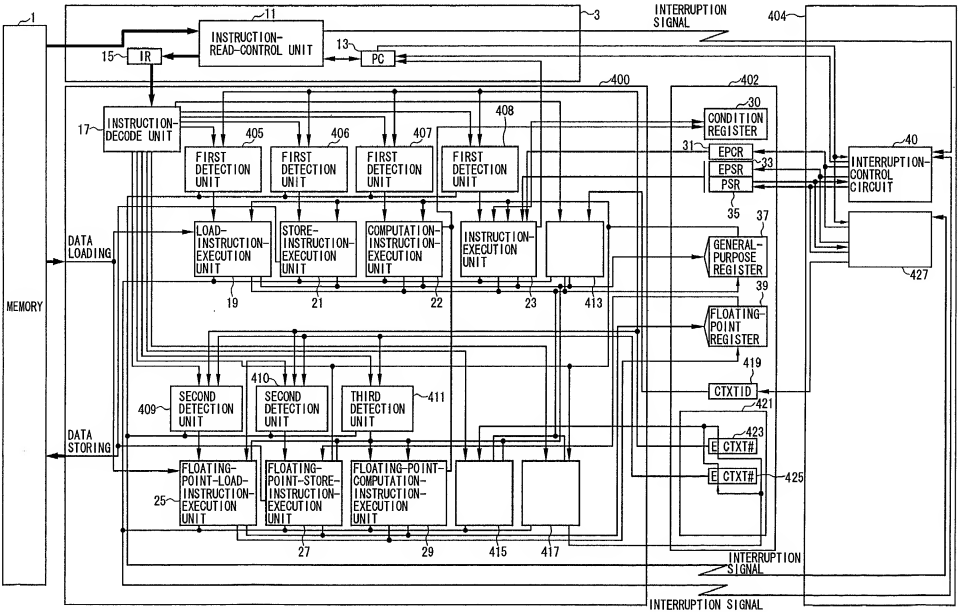

1
MEMORY
3
11
INSTRUCTION-READ-CONTROL UNIT
13
PC
15
IR
17
INSTRUCTION-DECODE UNIT
INTERRUPTION SIGNAL
404
400
402
30
CONDITION REGISTER
31
EPCR
33
EPSR
PSR
35
37
40
INTERRUPTION-CONTROL CIRCUIT
427
GENERAL-PURPOSE REGISTER
39
FLOATING-POINT REGISTER
419
CTXTID
421
423
E CTXT#
425
E CTXT#
405
406
407
408
FIRST DETECTION UNIT
FIRST DETECTION UNIT
FIRST DETECTION UNIT
FIRST DETECTION UNIT
DATA LOADING
LOAD-INSTRUCTION-EXECUTION UNIT
STORE-INSTRUCTION-EXECUTION UNIT
COMPUTATION-INSTRUCTION-EXECUTION UNIT
INSTRUCTION-EXECUTION UNIT
19
21
22
23
413
410
409
SECOND DETECTION UNIT
SECOND DETECTION UNIT
THIRD DETECTION UNIT
411
DATA STORING
25
FLOATING-POINT-LOAD-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-STORE-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-COMPUTATION-INSTRUCTION-EXECUTION UNIT
27
29
415
417
INTERRUPTION SIGNAL
INTERRUPTION SIGNAL

FIG.4

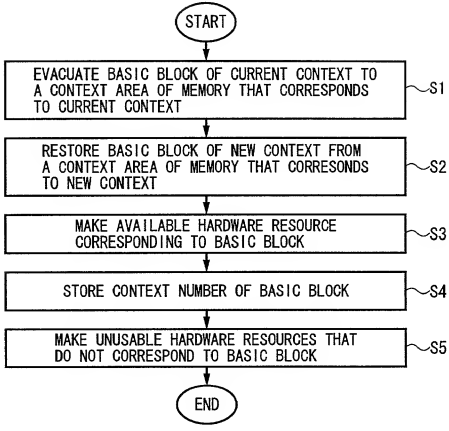

START
EVACUATE BASIC BLOCK OF CURRENT CONTEXT TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT
S1
RESTORE BASIC BLOCK OF NEW CONTEXT FROM A CONTEXT AREA OF MEMORY THAT CORRESONDS TO NEW CONTEXT
S2
MAKE AVAILABLE HARDWARE RESOURCE CORRESPONDING TO BASIC BLOCK
S3
STORE CONTEXT NUMBER OF BASIC BLOCK
S4
MAKE UNUSABLE HARDWARE RESOURCES THAT DO NOT CORRESPOND TO BASIC BLOCK
S5
END

FIG.5

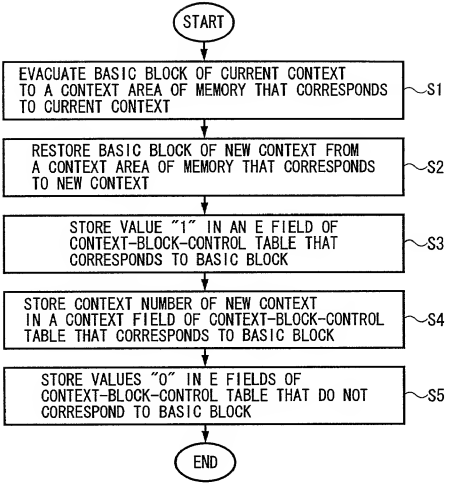

START
EVACUATE BASIC BLOCK OF CURRENT CONTEXT TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT
S1
RESTORE BASIC BLOCK OF NEW CONTEXT FROM A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO NEW CONTEXT
S2
STORE VALUE "1" IN AN E FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO BASIC BLOCK
S3
STORE CONTEXT NUMBER OF NEW CONTEXT IN A CONTEXT FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO BASIC BLOCK
S4
STORE VALUES "0" IN E FIELDS OF CONTEXT-BLOCK-CONTROL TABLE THAT DO NOT CORRESPOND TO BASIC BLOCK
S5
END

FIG.6

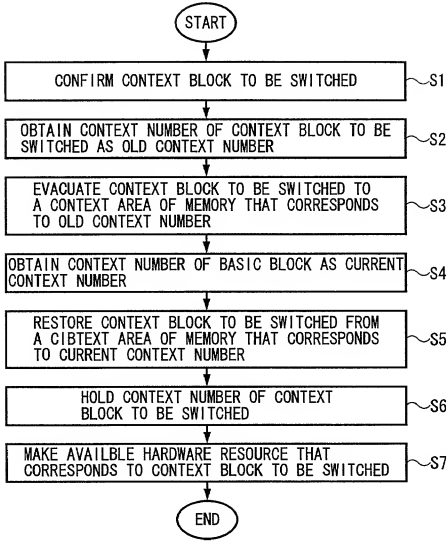

START
CONFIRM CONTEXT BLOCK TO BE SWITCHED
S1
OBTAIN CONTEXT NUMBER OF CONTEXT BLOCK TO BE SWITCHED AS OLD CONTEXT NUMBER
S2
EVACUATE CONTEXT BLOCK TO BE SWITCHED TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO OLD CONTEXT NUMBER
S3
OBTAIN CONTEXT NUMBER OF BASIC BLOCK AS CURRENT CONTEXT NUMBER
S4
RESTORE CONTEXT BLOCK TO BE SWITCHED FROM A CIBTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT NUMBER
S5
HOLD CONTEXT NUMBER OF CONTEXT BLOCK TO BE SWITCHED
S6
MAKE AVAILBLE HARDWARE RESOURCE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S7
END

FIG.7

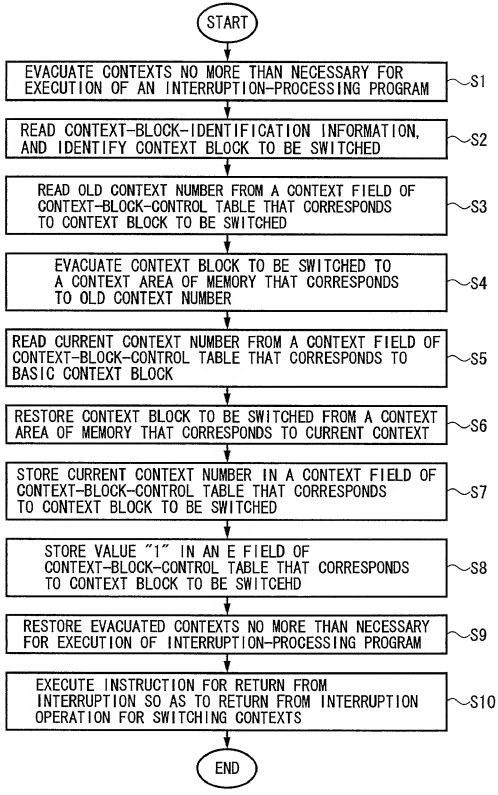

START
EVACUATE CONTEXTS NO MORE THAN NECESSARY FOR EXECUTION OF AN INTERRUPTION-PROCESSING PROGRAM
S1
READ CONTEXT-BLOCK-IDENTIFICATION INFORMATION, AND IDENTIFY CONTEXT BLOCK TO BE SWITCHED
S2
READ OLD CONTEXT NUMBER FROM A CONTEXT FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S3
EVACUATE CONTEXT BLOCK TO BE SWITCHED TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO OLD CONTEXT NUMBER
S4
READ CURRENT CONTEXT NUMBER FROM A CONTEXT FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO BASIC CONTEXT BLOCK
S5
RESTORE CONTEXT BLOCK TO BE SWITCHED FROM A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT
S6
STORE CURRENT CONTEXT NUMBER IN A CONTEXT FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S7
STORE VALUE "1" IN AN E FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCEHD
S8
RESTORE EVACUATED CONTEXTS NO MORE THAN NECESSARY FOR EXECUTION OF INTERRUPTION-PROCESSING PROGRAM
S9
EXECUTE INSTRUCTION FOR RETURN FROM INTERRUPTION SO AS TO RETURN FROM INTERRUPTION OPERATION FOR SWITCHING CONTEXTS
S10
END

FIG.8

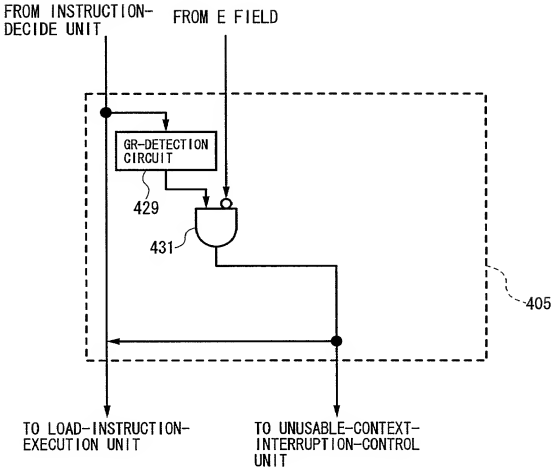

FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
GR-DETECTION CIRCUIT
429
431
405
TO LOAD-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

FIG.9

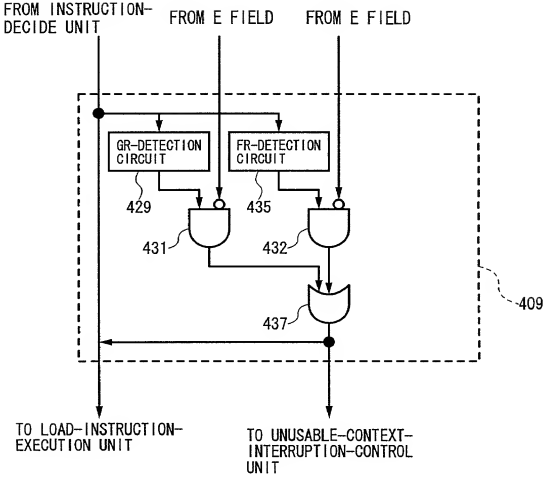
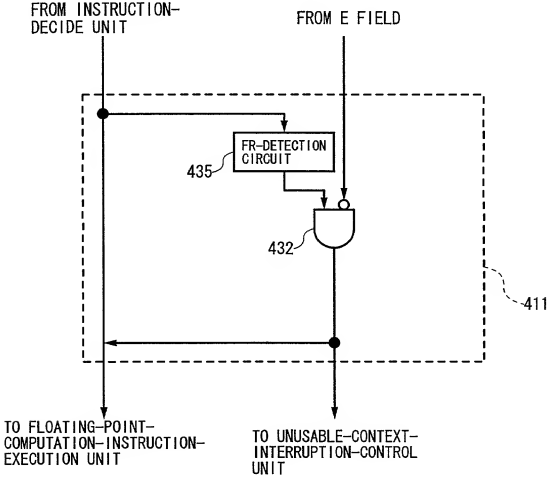

FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FROM E FIELD
GR-DETECTION CIRCUIT
FR-DETECTION CIRCUIT
429
435
431
432
437
409
TO LOAD-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

FIG.10


FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FR-DETECTION CIRCUIT
435
432
411
TO FLOATING-POINT-COMPUTATION-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

TRANSLATION ONLY - NOT TO SCALE

FIG.11

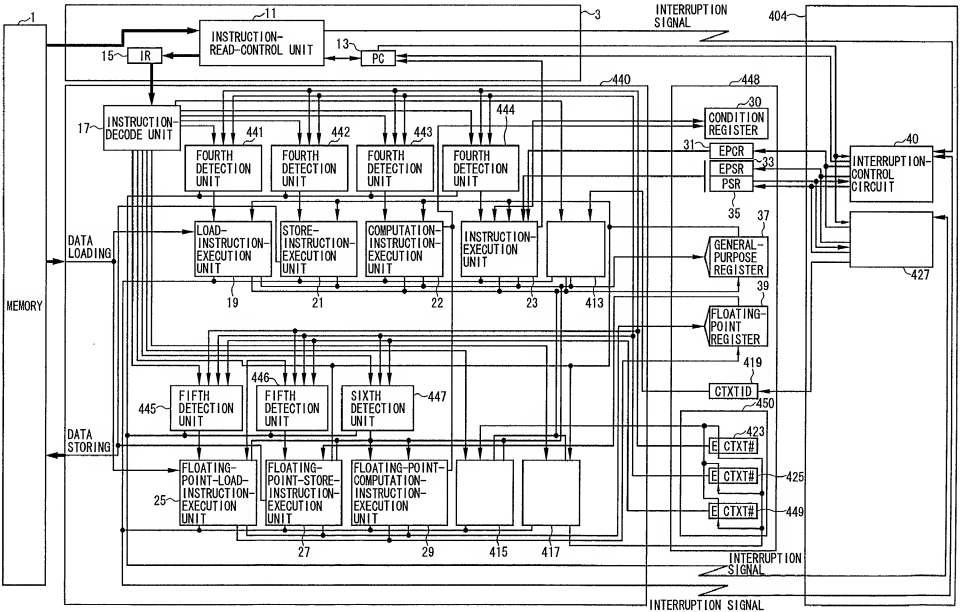


FIG.12

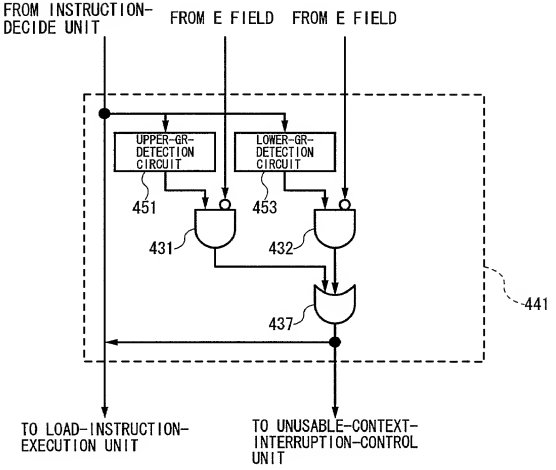

FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FROM E FIELD
UPPER-GR-DETECTION CIRCUIT
LOWER-GR-DETECTION CIRCUIT
451
453
431
432
437
441
TO LOAD-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

FIG.13

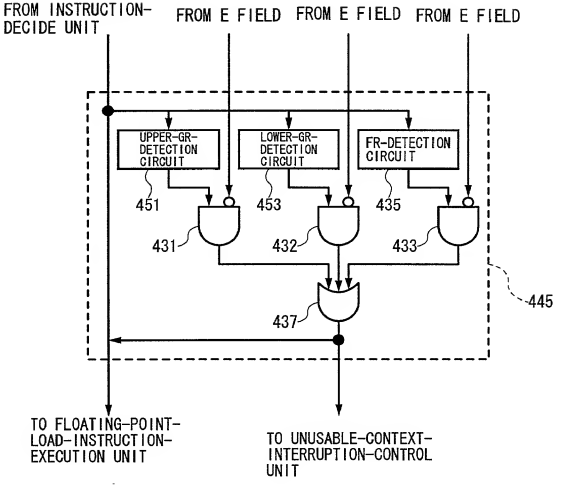
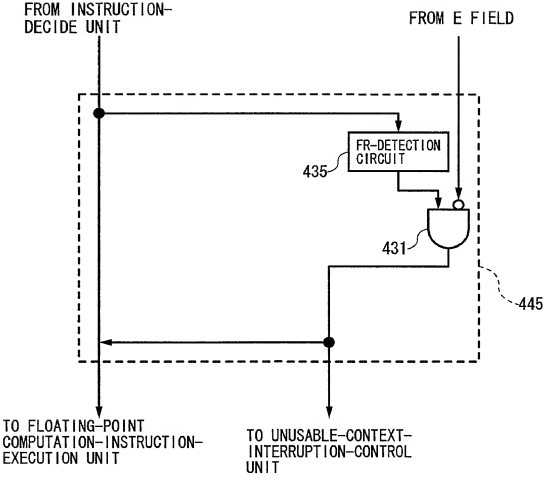

FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FROM E FIELD
FROM E FIELD
UPPER-GR-DETECTION CIRCUIT
LOWER-GR-DETECTION CIRCUIT
FR-DETECTION CIRCUIT
451
453
435
431
432
433
437
445
TO FLOATING-POINT-LOAD-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

FIG.14


FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FR-DETECTION CIRCUIT
435
431
445
TO FLOATING-POINT COMPUTATION-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

TOSHIBA 06-0524-0

FIG.15

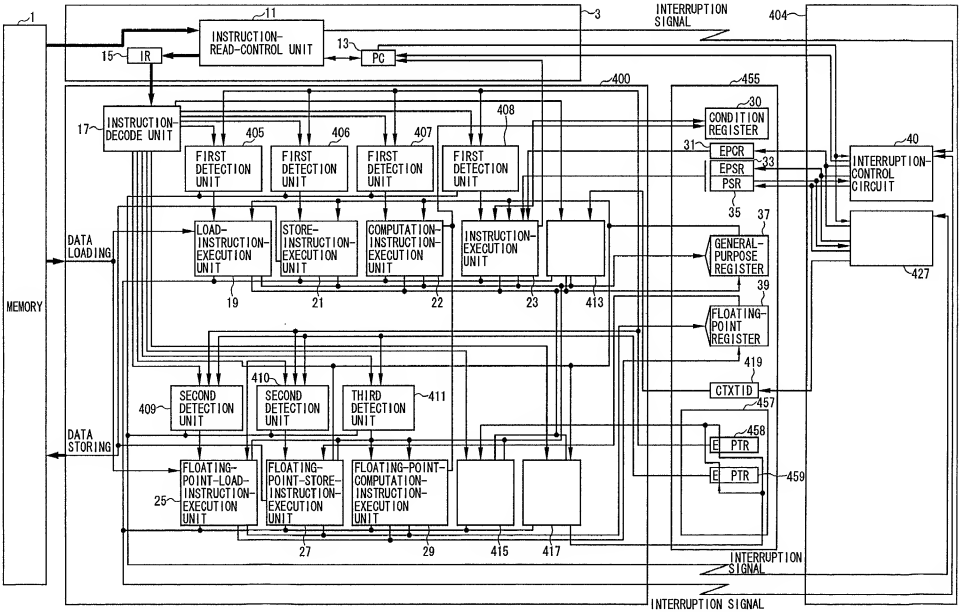


FIG.16

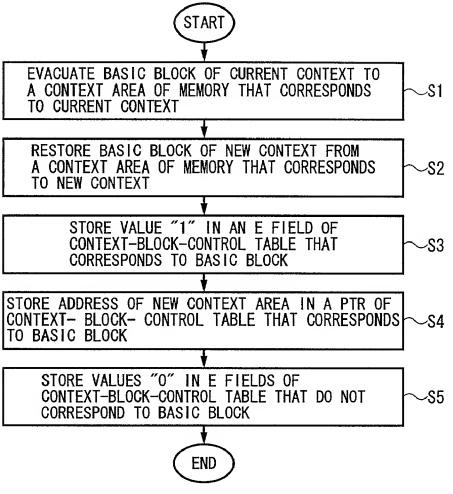
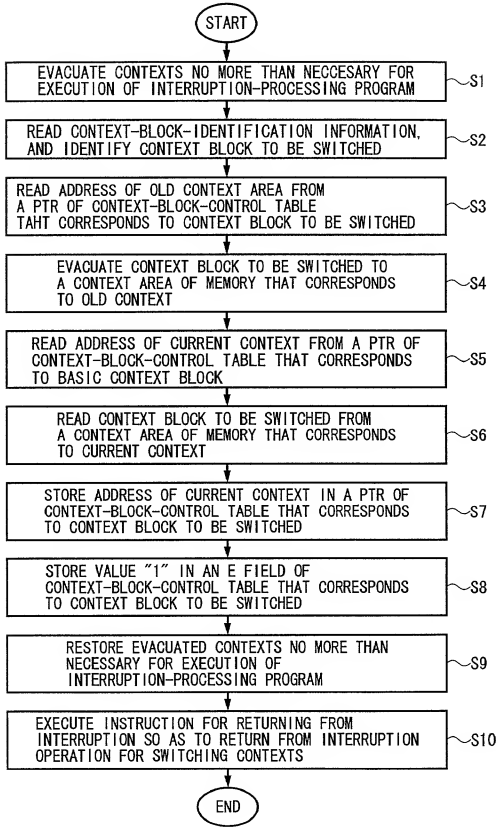

START
EVACUATE BASIC BLOCK OF CURRENT CONTEXT TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT
S1
RESTORE BASIC BLOCK OF NEW CONTEXT FROM A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO NEW CONTEXT
S2
STORE VALUE "1" IN AN E FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO BASIC BLOCK
S3
STORE ADDRESS OF NEW CONTEXT AREA IN A PTR OF CONTEXT- BLOCK- CONTROL TABLE THAT CORRESPONDS TO BASIC BLOCK
S4
STORE VALUES "0" IN E FIELDS OF CONTEXT-BLOCK-CONTROL TABLE THAT DO NOT CORRESPOND TO BASIC BLOCK
S5
END

FIG.17


START
EVACUATE CONTEXTS NO MORE THAN NECCESARY FOR EXECUTION OF INTERRUPTION-PROCESSING PROGRAM
S1
READ CONTEXT-BLOCK-IDENTIFICATION INFORMATION, AND IDENTIFY CONTEXT BLOCK TO BE SWITCHED
S2
READ ADDRESS OF OLD CONTEXT AREA FROM A PTR OF CONTEXT-BLOCK-CONTROL TABLE TAHT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S3
EVACUATE CONTEXT BLOCK TO BE SWITCHED TO A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO OLD CONTEXT
S4
READ ADDRESS OF CURRENT CONTEXT FROM A PTR OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO BASIC CONTEXT BLOCK
S5
READ CONTEXT BLOCK TO BE SWITCHED FROM A CONTEXT AREA OF MEMORY THAT CORRESPONDS TO CURRENT CONTEXT
S6
STORE ADDRESS OF CURRENT CONTEXT IN A PTR OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S7
STORE VALUE "1" IN AN E FIELD OF CONTEXT-BLOCK-CONTROL TABLE THAT CORRESPONDS TO CONTEXT BLOCK TO BE SWITCHED
S8
RESTORE EVACUATED CONTEXTS NO MORE THAN NECESSARY FOR EXECUTION OF INTERRUPTION-PROCESSING PROGRAM
S9
EXECUTE INSTRUCTION FOR RETURNING FROM INTERRUPTION SO AS TO RETURN FROM INTERRUPTION OPERATION FOR SWITCHING CONTEXTS
S10
END

PURSUANT TO 37 CFR 1.84(h)(2)

FIG.18

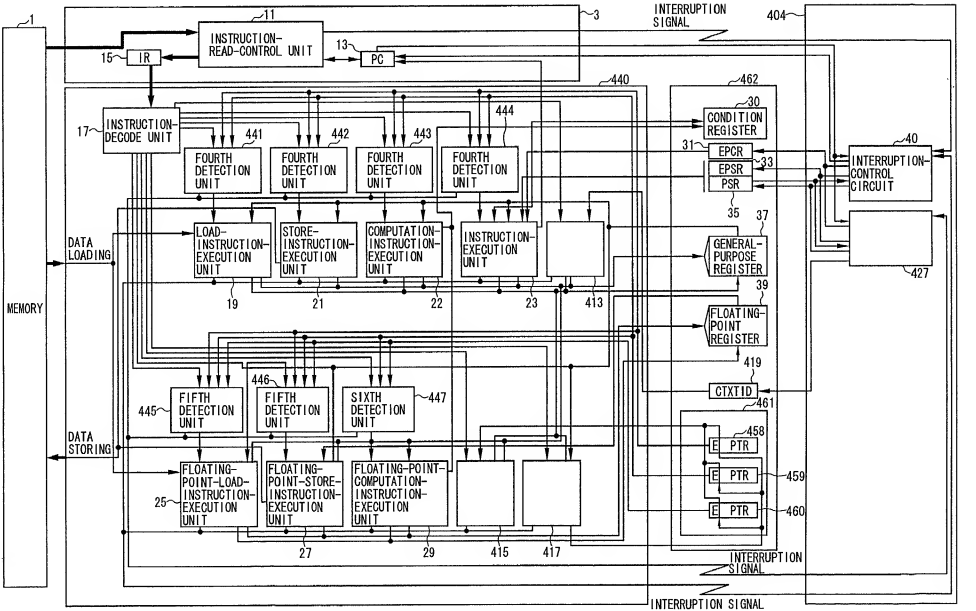


FIG.19

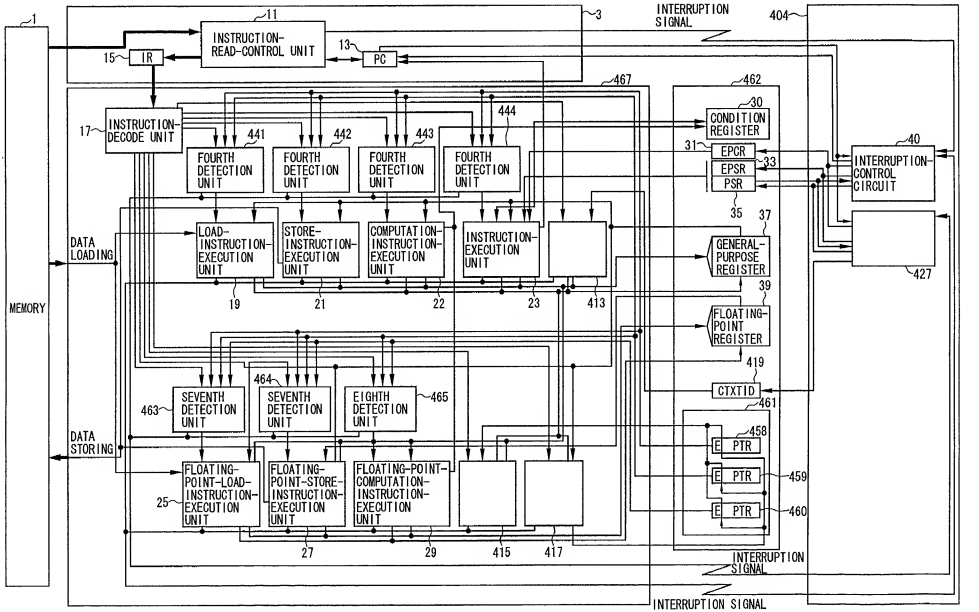

1
MEMORY
11
INSTRUCTION-READ-CONTROL UNIT
15
IR
13
PC
3
INTERRUPTION SIGNAL
404
17
INSTRUCTION-DECODE UNIT
441
442
443
444
FOURTH DETECTION UNIT
FOURTH DETECTION UNIT
FOURTH DETECTION UNIT
FOURTH DETECTION UNIT
467
462
30
CONDITION REGISTER
31
EPCR
33
EPSR
PSR
35
37
40
INTERRUPTION-CONTROL CIRCUIT
427
DATA LOADING
LOAD-INSTRUCTION-EXECUTION UNIT
STORE-INSTRUCTION-EXECUTION UNIT
COMPUTATION-INSTRUCTION-EXECUTION UNIT
INSTRUCTION-EXECUTION UNIT
GENERAL-PURPOSE REGISTER
39
FLOATING-POINT REGISTER
19
21
22
23
413
419
CTXTID
461
458
E PTR
459
E PTR
460
E PTR
464
463
SEVENTH DETECTION UNIT
SEVENTH DETECTION UNIT
EIGHTH DETECTION UNIT
465
DATA STORING
25
FLOATING-POINT-LOAD-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-STORE-INSTRUCTION-EXECUTION UNIT
FLOATING-POINT-COMPUTATION-INSTRUCTION-EXECUTION UNIT
27
29
415
417
INTERRUPTION SIGNAL
INTERRUPTION SIGNAL

FIG.20

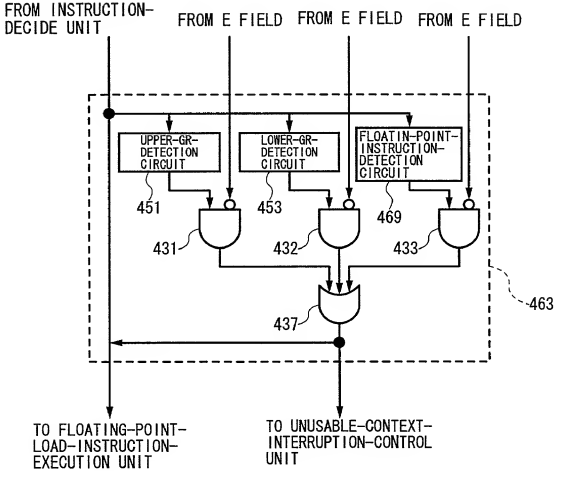
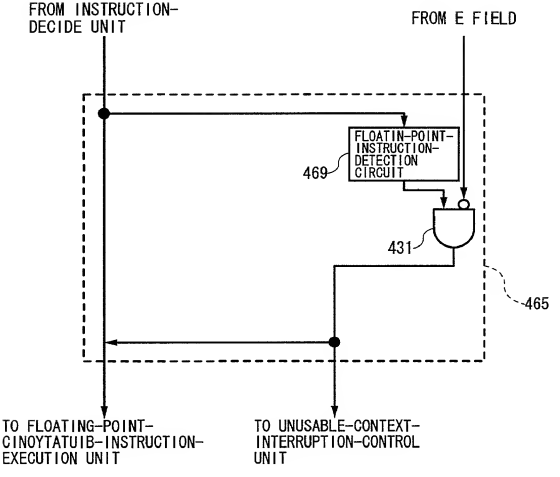

FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FROM E FIELD
FROM E FIELD
UPPER-GR-DETECTION CIRCUIT
LOWER-GR-DETECTION CIRCUIT
FLOATIN-POINT-INSTRUCTION-DETECTION CIRCUIT
451
453
469
431
432
433
437
463
TO FLOATING-POINT-LOAD-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

FIG.21


FROM INSTRUCTION-DECIDE UNIT
FROM E FIELD
FLOATIN-POINT-INSTRUCTION-DETECTION CIRCUIT
469
431
465
TO FLOATING-POINT-CINOYTATUIB-INSTRUCTION-EXECUTION UNIT
TO UNUSABLE-CONTEXT-INTERRUPTION-CONTROL UNIT

TRANSLATION DRAWING

FIG.22

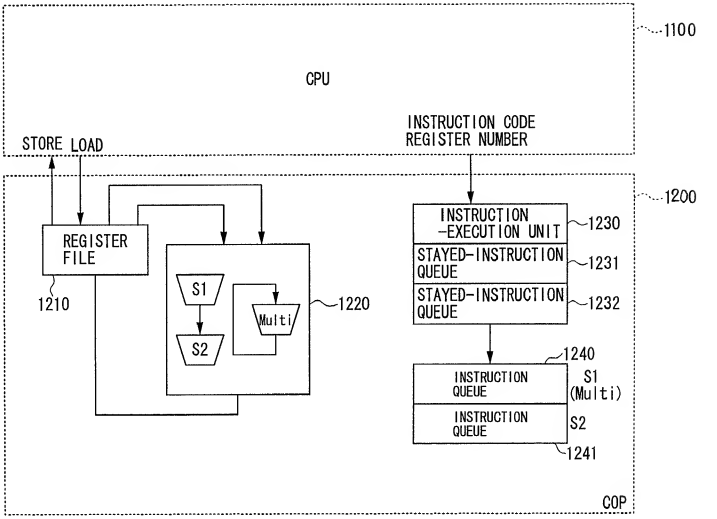


FIG.23

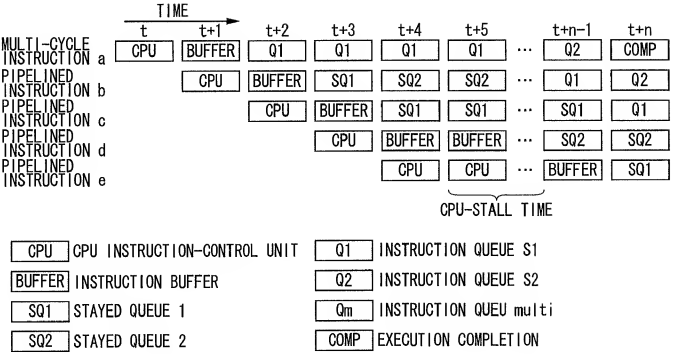

TIME
t
t+1
t+2
t+3
t+4
t+5
t+n-1
t+n
MULTI-CYCLE INSTRUCTION a
PIPELINED INSTRUCTION b
PIPELINED INSTRUCTION c
PIPELINED INSTRUCTION d
PIPELINED INSTRUCTION e
CPU
BUFFER
Q1
SQ1
SQ2
Q2
COMP
CPU-STALL TIME
CPU CPU INSTRUCTION-CONTROL UNIT
BUFFER INSTRUCTION BUFFER
SQ1 STAYED QUEUE 1
SQ2 STAYED QUEUE 2
Q1 INSTRUCTION QUEUE S1
Q2 INSTRUCTION QUEUE S2
Qm INSTRUCTION QUEU multi
COMP EXECUTION COMPLETION

FIG.24

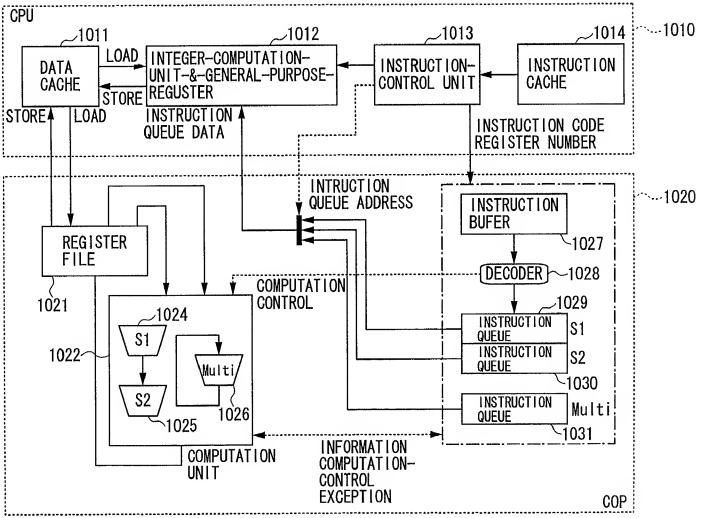

CPU
1011
DATA CACHE
LOAD
STORE
1012
INTEGER-COMPUTATION-UNIT-&-GENERAL-PURPOSE-REGUSTER
1013
INSTRUCTION-CONTROL UNIT
1014
INSTRUCTION CACHE
1010
STORE
LOAD
INSTRUCTION QUEUE DATA
INSTRUCTION CODE REGISTER NUMBER
INTRUCTION QUEUE ADDRESS
1020
INSTRUCTION BUFER
1027
DECODER
1028
1029
REGISTER FILE
1021
COMPUTATION CONTROL
1024
S1
1022
Multi
S2
1025
1026
INSTRUCTION QUEUE
S1
INSTRUCTION QUEUE
S2
1030
INSTRUCTION QUEUE
Multi
1031
COMPUTATION UNIT
INFORMATION COMPUTATION-CONTROL EXCEPTION
COP

FIG.25

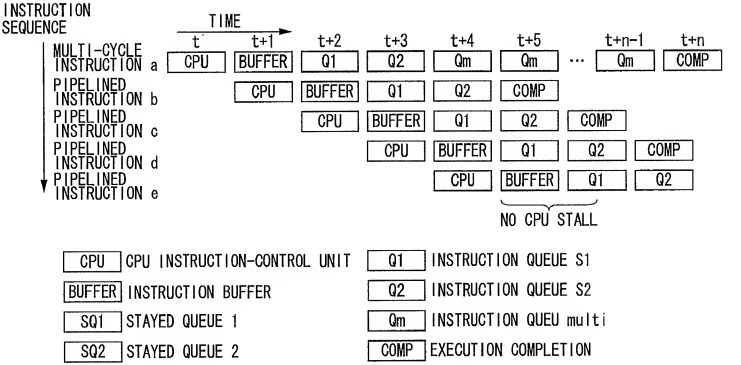

INSTRUCTION SEQUENCE
TIME
t
t+1
t+2
t+3
t+4
t+5
t+n-1
t+n
MULTI-CYCLE INSTRUCTION a
PIPELINED INSTRUCTION b
PIPELINED INSTRUCTION c
PIPELINED INSTRUCTION d
PIPELINED INSTRUCTION e
CPU
BUFFER
Q1
Q2
Qm
COMP
NO CPU STALL
CPU CPU INSTRUCTION-CONTROL UNIT
BUFFER INSTRUCTION BUFFER
SQ1 STAYED QUEUE 1
SQ2 STAYED QUEUE 2
Q1 INSTRUCTION QUEUE S1
Q2 INSTRUCTION QUEUE S2
Qm INSTRUCTION QUEU multi
COMP EXECUTION COMPLETION

FIG.26

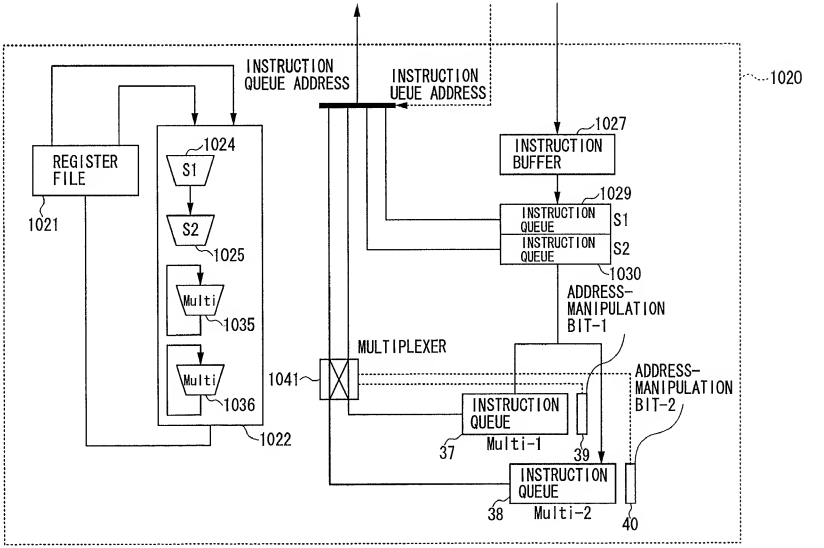

INSTRUCTION QUEUE ADDRESS
INSTRUCTION UEUE ADDRESS
1020
REGISTER FILE
1021
1024
S1
S2
1025
Multi
1035
Multi
1036
1022
1027
INSTRUCTION BUFFER
1029
INSTRUCTION QUEUE
S1
INSTRUCTION QUEUE
S2
1030
ADDRESS-MANIPULATION BIT-1
MULTIPLEXER
1041
ADDRESS-MANIPULATION BIT-2
INSTRUCTION QUEUE
Multi-1
37
39
INSTRUCTION QUEUE
38
Multi-2
40

FIG.27

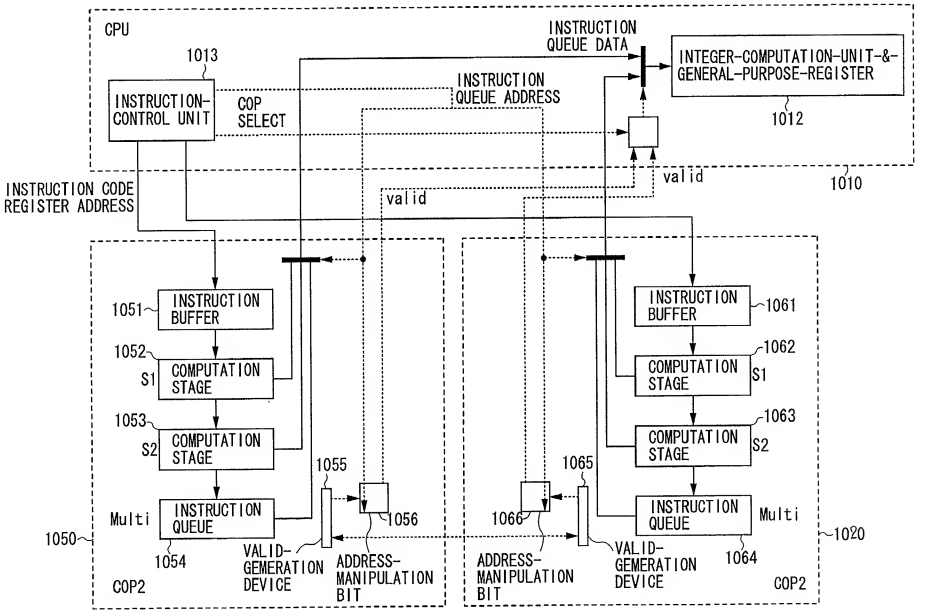

CPU
INSTRUCTION QUEUE DATA
INTEGER-COMPUTATION-UNIT-&-GENERAL-PURPOSE-REGISTER
1012
1013
INSTRUCTION-CONTROL UNIT
COP SELECT
INSTRUCTION QUEUE ADDRESS
INSTRUCTION CODE REGISTER ADDRESS
valid
valid
1010
1051
INSTRUCTION BUFFER
1052
S1
COMPUTATION STAGE
1053
S2
COMPUTATION STAGE
Multi
INSTRUCTION QUEUE
1050
1054
1055
1056
VALID-GEMERATION DEVICE
ADDRESS-MANIPULATION BIT
COP2
1061
INSTRUCTION BUFFER
1062
S1
COMPUTATION STAGE
1063
S2
COMPUTATION STAGE
1065
1066
Multi
INSTRUCTION QUEUE
1064
ADDRESS-MANIPULATION BIT
VALID-GEMERATION DEVICE
COP2
1020

FIG. 28

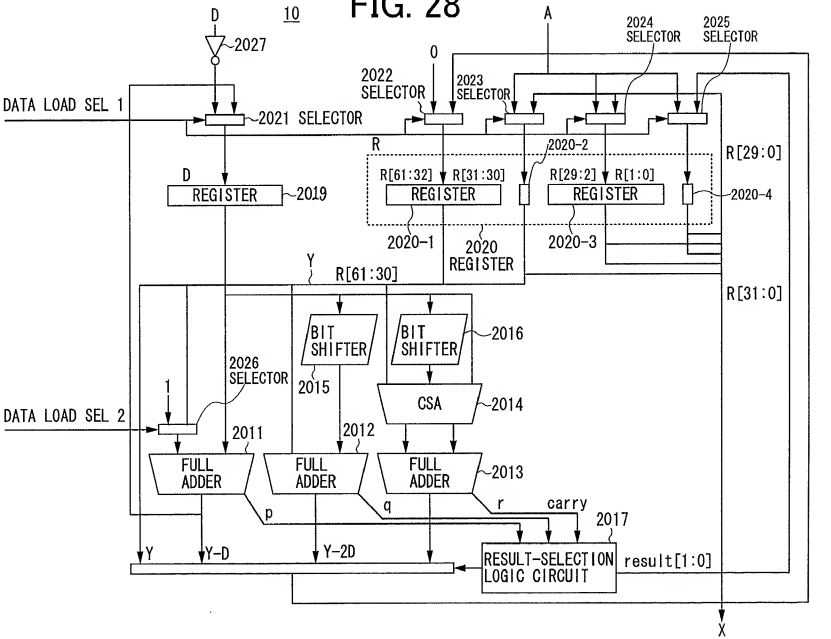
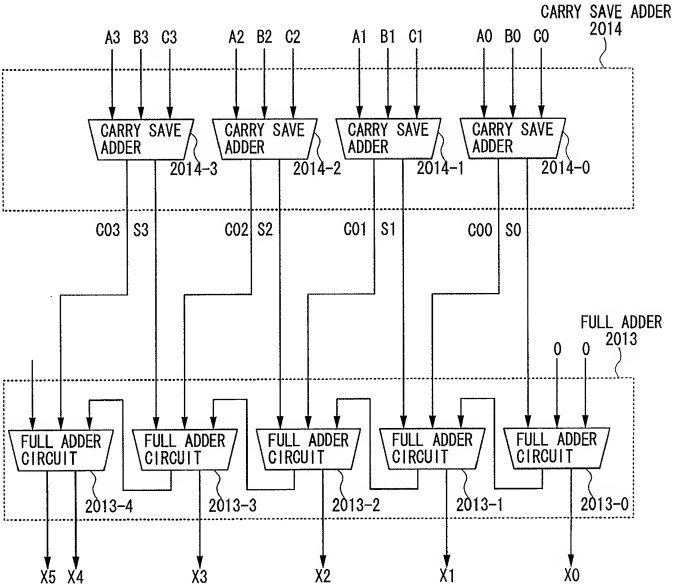

10
D
2027
A
2024 SELECTOR
2025 SELECTOR
0
2022 SELECTOR
2023 SELECTOR
DATA LOAD SEL 1
2021 SELECTOR
R
2020-2
R[29:0]
D
REGISTER
2019
R[61:32]
R[31:30]
REGISTER
R[29:2]
R[1:0]
REGISTER
2020-4
2020-1
2020 REGISTER
2020-3
Y
R[61:30]
R[31:0]
BIT SHIFTER
BIT SHIFTER
2016
2026 SELECTOR
1
2015
CSA
2014
DATA LOAD SEL 2
2011
2012
FULL ADDER
FULL ADDER
FULL ADDER
2013
p
q
r
carry
2017
Y
Y-D
Y-2D
RESULT-SELECTION LOGIC CIRCUIT
result[1:0]
X

FIG.29

INPUT			QUOTIENT (result)	SELECTED REMAINDER
p	q	r		
0	0	0	00	Y
1	0	0	01	Y-D
X	1	0	10	Y-2D
X	X	1	11	Y-3D

FIG.30


CARRY SAVE ADDER
2014
A3 B3 C3
A2 B2 C2
A1 B1 C1
A0 B0 C0
CARRY SAVE ADDER
2014-3
CARRY SAVE ADDER
2014-2
CARRY SAVE ADDER
2014-1
CARRY SAVE ADDER
2014-0
C03 S3
C02 S2
C01 S1
C00 S0
FULL ADDER
2013
0 0
FULL ADDER CIRCUIT
2013-4
FULL ADDER CIRCUIT
2013-3
FULL ADDER CIRCUIT
2013-2
FULL ADDER CIRCUIT
2013-1
FULL ADDER CIRCUIT
2013-0
X5 X4
X3
X2
X1
X0

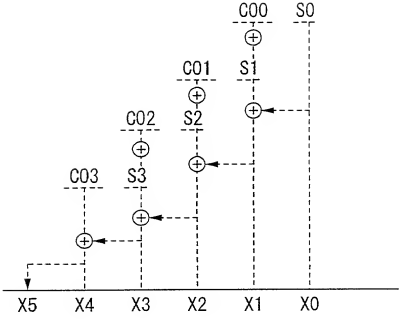



FIG. 31